Numerical Optimization - Syllabus.

Bart G. van Bloemen Waanders

August 16, 2005

1 General Description

Computational science, engineering and applied mathematics face a growing need to develop algorithms, methods, and simulation codes that solve difficult and large scale problems. Furthermore, solutions are desired that can provide designs, controls, and inversion results for the best choice of input parameters. Manually perturbing these design spaces is difficult when the problem is constrained by dynamical systems, and nearly impossible if combined with large number of design variables. Numerical optimization algorithms can provide computer scientist, engineers and mathematicians an avenue to 1) provide the most desirable solution, 2) automate the execution, and 3) achieve efficient convergence rates.

This graduate level class will provide the theoretical and practical foundation for numerical optimization. Algorithms for gradient based algorithms will be presented in addition to a basic review of non-gradient approaches. Constrained and unconstrained formulations with appropriate globalization methods including line search and trust region methods will be applied to interesting problems. Several solution methods will be discussed such as conjugate gradient, Newton and Quasi Newton algorithms, nonlinear least squares, quadratic programming and sequential quadratic programming.

The focus of the class will be to build up a theoretical foundation combined with practical applications and computational implementation strategies. A variety of dynamical systems will be used as application examples and provide the basis for projects. Longer term coding projects will be assigned using popular computational languages and tools.

2 Course Outline

1. Introduction - general concepts, simple examples, global vs local, gradient vs nongradient black box vs SAND methods - week 1

2. Unconstrained optimization - overview of algorithms, globalization issues - week 2-3
3. Globalization - line search methods, trust region methods - week 4-5
4. Conjugate Gradient - Linear, Nonlinear, basic properties - week 6
5. Newton Methods - Hessian calculation, Line search Newton, trust region Newton, convergence issues, quasi-newton, BFGS, SR1, Broyden, limited memory BFGS, Gauss Newton Methods, Nonlinear issues - week 7-9
6. Constrained Optimization - First and second order optimality conditions, nonlinear case - week 10-11
7. Quadratic Programming - Range and null space methods, active set methods for inequalities, penalty methods, sequential quadratic programming - week 12 - 15
8. Review - week 16

3 Projects, Assignments, Format

Numerical optimization requires an understanding of algorithms, implementation, and problem solving. To meet those goals, approximately two week projects will be assigned consisting of algorithm implementation and solving specific problems. At the conclusion of the class, students will have developed a comprehensive collection of primarily gradient based optimization tools and will have solved a variety of problems. These two week assignments will consist of implementation of algorithms that have been discussed during class lectures. The coding language will be flexible (C++, C, Java, Matlab). Homework assignments will supplement the process of understanding these algorithms. One of the key goals of the two week assignments is to translate mathematics into efficient coding implementations and provide an opportunity to apply algorithms to a range of problems.

4 Schedule, Prerequisites, Grading, and Texts

The class will be scheduled for the Fall 05 semester, consisting of 16 weeks, approximately 32 lectures, 1.5 hours per lecture. Prerequisites for the class consist of calculus, basic linear algebra and numerical methods, and some programming skills. The recommended text is “Numerical Optimization” by J. Nocedal and S. Wright. A supporting text will be “Practical Optimization” P. Gill, W. Murray, and M. Wright.